PATENT ABSTRACTS OF JAPAN

(11)Publication number : **2001-326935**
(43)Date of publication of application : **22.11.2001**

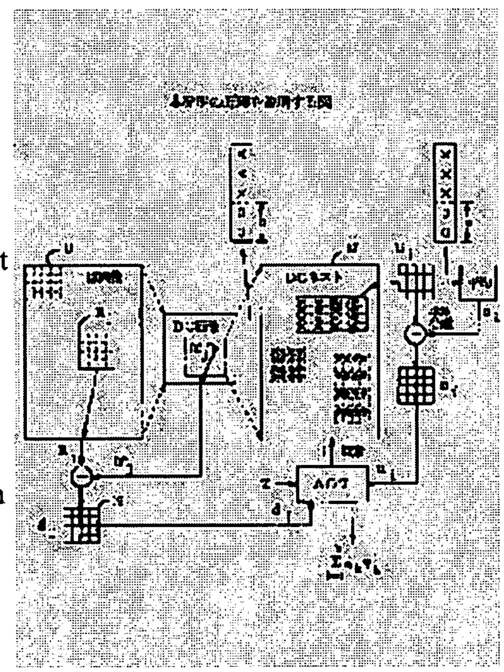
(51)Int.CI. H04N 7/30
H03M 7/30

(21)Application number : **2000-141675**
(22)Date of filing : **15.05.2000**

(71)Applicant : **HUDSON SOFT CO LTD**
(72)Inventor : **ITAGAKI FUMIHIKO**
KAWASHIMA MIYUKI

(54) IMAGE CODING/DECODING METHOD, ITS DEVICE, AND RECORDING MEDIUM RECORDED WITH PROGRAM THEREFOR

(57)Abstract:
PROBLEM TO BE SOLVED: To obtain high image quality and high-speed coding/decoding in an image coding/decoding method, its device, and a recording medium in which its program is recorded.
SOLUTION: In an image coding method to generate a DC image consisting of each block average obtained by dividing an image data into blocks for every B pixel for making the part into a DC nest, and to obtain one or two or more orthogonal bases (αk <vk> or the like) for approximating the residual vector <dj> by an accomodative orthogonal transformation (AOT) using DC nest when the size of a residual vector <dj> obtained by separating a DC value DCJ from a pixel block <Rj> for coding exceeds an allowable value Z, each inferior n bit (n=log2B) of a base candidate block <Ui> to which a down sample is applied from the DC nest is set as 0. Besides, the block average ai is separated from the base candidate block <Ui> to generate a base candidate vector <ui>.



(19)日本国特許庁（ＪＰ）　(12)**公開特許公報**（Ａ）　(11)特許出願公開番号
特開2001－326935
（P2001－326935A）
(43)公開日　平成13年11月22日(2001.11.22)

(51)Int.Cl.⁷	識別記号	ＦＩ		テーマコード゛(参考)
Ｈ０４Ｎ　7/30		Ｈ０３Ｍ　7/30	Ａ	５Ｃ０５９
Ｈ０３Ｍ　7/30			Ｂ	５Ｊ０６４
		Ｈ０４Ｎ　7/133	Ｚ	

審査請求　未請求　請求項の数16　ＯＬ　（全 33 頁）

(21)出願番号　特願2000－141675(P2000－141675)
(22)出願日　平成12年５月15日(2000.5.15)

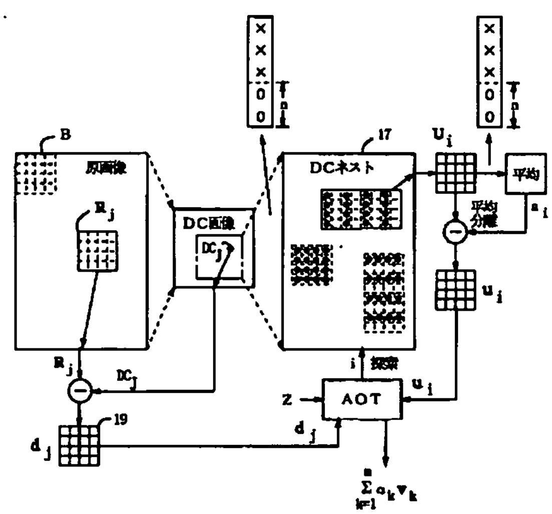
(71)出願人　591095856
株式会社ハドソン
北海道札幌市豊平区平岸三条五丁目４番22号
(72)発明者　板垣　史彦
北海道札幌市南区芸術の森３丁目Ｃ62　株式会社ハドソン内
(72)発明者　川島　深雪
北海道札幌市南区芸術の森３丁目Ｃ62　株式会社ハドソン内
(74)代理人　100097087
弁理士　▲高▼須　宏

最終頁に続く

(54)【発明の名称】　画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体

(57)【要約】
【課題】　画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体に関し、高画質かつ高速の符号／復号が得られることを課題とする。
【解決手段】　画像データをＢ画素毎にブロック分割して各ブロック平均値からなるＤＣ画像を生成し、その一部をＤＣネストとすると共に、符号対象の画素ブロック〈R_j〉からそのＤＣ値DC_Jを分離した残差ベクトル〈d_j〉の大きさが許容値Ｚを超える場合は、残差ベクトル〈d_j〉を近似するための１又は２以上の直交基底（α_k〈v_k〉等）をＤＣネストを使用した適応的直交変換（ＡＯＴ）により求める画像符号方法において、ＤＣネストからダウンサンプルされた基底候補ブロック〈U_i〉の各下位ｎ（$n = \log_2 B$）ビットが０にされている。またこの基底候補ブロック〈U_i〉からそのブロック平均値a_iを分離して基底候補ベクトル〈u_i〉を生成する。

本発明の原理を説明する図



【特許請求の範囲】

【請求項１】　画像データをＢ画素毎にブロック分割して各ブロック平均値からなるＤＣ画像を生成し、その一部をＤＣネストとすると共に、符号対象の画素ブロックからそのＤＣ値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための１又は２以上の直交基底をＤＣネストを使用した適応的直交変換により求める画像符号方法において、

ＤＣネストから基底候補ブロックをダウンサンプルしてそのブロック平均値を求める際の各サンプルＤＣ画素の下位n（$n=\log_2 B$）ビットが０にされていることを特徴とする画像符号方法。

【請求項２】　ＤＣ画像からＤＣネストを生成する際に各ＤＣ画素の下位ｎビットを０にすることを特徴とする請求項１に記載の画像符号方法。

【請求項３】　下位ｎビットが０にされている基底候補ブロックからそのブロック平均値を分離して残差ベクトルを近似するための基底候補ベクトルを生成することを特徴とする請求項１又は２に記載の画像符号方法。

【請求項４】　基底候補ベクトル$\langle u_i \rangle$の任意要素（例えばu_{16}）を残りの要素の一次結合で置き換えると共に、該基底候補ベクトル$\langle u_i \rangle$と任意他のベクトル$\langle w \rangle$との内積を、

$$\langle w \cdot u_i \rangle = (w_1 - w_{16})\, u_1 + (w_2 - w_{16})\, u_2 +, \cdots, + (w_{15} - w_{16})\, u_{15}$$

の積和演算により求めることを特徴とする請求項３に記載の画像符号方法。

【請求項５】　残差ベクトル$\langle d \rangle$、基底候補ベクトル$\langle u_i \rangle$とするときに第１基底の探索は、

$$h_i = \langle d \cdot u_i \rangle^2 / \| u_i \|^2$$

を最大とするものを条件に探索することを特徴とする請求項３又は４に記載の画像符号方法。

【請求項６】　残差ベクトル$\langle d \rangle$、第１基底に対応する基底候補ベクトル$\langle u_1 \rangle$、第２基底を探索する基底候補ベクトル$\langle u_i \rangle$とするときに第２基底の探索は、

$$h_i = \{ \langle d \cdot u_i \rangle - (\langle d \cdot u_1 \rangle \langle u_1 \cdot u_i \rangle) / \| u_1 \|^2 \}^2 / \{ \| u_i \|^2 - (\langle u_1 \cdot u_i \rangle) / \| u_1 \|)^2 \}$$

を最大とするものを条件に探索することを特徴とする請求項３又は４に記載の画像符号方法。

【請求項７】　残差ベクトル$\langle d \rangle$、第１正規基底ベクトル$\langle v_1 \rangle$、第２正規直交基底ベクトル$\langle v_2 \rangle$、第３基底を探索する基底候補ベクトル$\langle u_i \rangle$とするときに第３基底の探索は、

$$h_i = (\langle d \cdot u_i \rangle - (d \cdot v_1 \rangle \langle v_1 \cdot u_i \rangle - \langle d \cdot v_2 \rangle \langle v_2 \cdot u_i \rangle)^2 / \{ \| u_i \|^2 - \langle v_1 \cdot u_i \rangle^2 - \langle v_2 \cdot u_i \rangle^2 \}$$

を最大とするものを条件に探索することを特徴とする請求項３又は４に記載の画像符号方法。

【請求項８】　探索条件にマッチした基底候補ベクトル$\langle u_i \rangle$をそれ以前の１又は２以上の正規直交基底に正規直交化することを特徴とする請求項６又は７に記載の画像符号方法。

【請求項９】　画像データをＢ画素毎にブロック分割して各ブロック平均値からなるＤＣ画像を生成し、その一部をＤＣネストとすると共に、符号対象の画素ブロックからそのＤＣ値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための１又は２以上の直交基底をＤＣネストを使用した適応的直交変換により求める画像符号方法において、

求められた基底系を$\beta_k \langle u_k \rangle$（$k=1 \sim m$）とするときに、各スカラー展開係数$\beta_1 \sim \beta_m$のノルムをその大きさ順に並べ替え、０を含む隣接ノルム間の各差分を求め、得られた各差分につきハフマン符号を適用することを特徴とする画像符号方法。

【請求項１０】　画像データをＢ画素毎にブロック分割して各ブロック平均値からなるＤＣ画像を生成し、その一部をＤＣネストとすると共に、符号対象の画素ブロックからそのＤＣ値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための１又は２以上の直交基底をＤＣネストを使用した適応的直交変換により求める画像符号方法において、

求めた基底数が所定以上の場合は基底系の符号化に代えて、符号対象ブロックの画像データそのものを符号化することを特徴とする画像符号方法。

【請求項１１】　ＨＶＱ方式に係る符号データからＢ画素毎の各ブロック平均値に相当するＤＣ画像を再生し、その一部をＤＣネストとすると共に、ターゲットブロックのＤＣ値に対し、前記符号データに基づきＤＣネストから選択生成した１又は２以上の基底ベクトルを合成してターゲットブロックの画像データを再生する画像復号方法において、

ＤＣネストから選択ブロックをダウンサンプルしてそのブロック平均値を求める際の各サンプルＤＣ画素の下位n（$n=\log_2 B$）ビットが０にされていることを特徴とする画像復号方法。

【請求項１２】　ＨＶＱ方式に係る符号データからＢ画素毎の各ブロック平均値に相当するＤＣ画像を再生し、その一部をＤＣネストとすると共に、ターゲットブロックのＤＣ値に対し、前記符号データに基づきＤＣネストから選択生成した１又は２以上の基底ベクトルを合成してターゲットブロックの画像データを再生する画像復号方法において、

復号された基底系が$\beta_k \langle u_k \rangle$（$k=1 \sim m$）に係る情報であるときに、ＤＣネストから読み出した各選択ブロック$\langle U_k \rangle$につき各ＤＣ画素の下位n（$n=\log_2 B$）ビットが０にされていると共に、まず$\beta_k \langle U_k \rangle$（$k=1 \sim m$）の積和演算を行い、その演算結果をブロック画素数Ｂで除算することを特徴とする画像復号方法。

【請求項１３】　ＤＣ画像からＤＣネストを生成する際

に各ＤＣ画素の下位ｎビットを０にすることを特徴とする請求項１１又は１２に記載の画像復号方法。

【請求項１４】　画像データをＢ画素毎にブロック分割して各ブロック平均値からなるＤＣ画像を生成し、その一部をＤＣネストとすると共に、符号対象の画素ブロックからそのＤＣ値を分離した残差ベクトルの大きさが許容値を超える場合は、該残差ベクトルを近似するための１又は２以上の直交基底をＤＣネストを使用した適応的直交変換により求める画像符号装置において、

各ＤＣネスト画素の下位ｎ（$n=\log_2 B$）ビットが０にされているＤＣネストを記憶するメモリを備えることを特徴とする画像符号装置。

【請求項１５】　ＨＶＱ方式に係る符号データからＢ画素毎の各ブロック平均値に相当するＤＣ画像を再生し、その一部をＤＣネストとすると共に、ターゲットブロックのＤＣ値に対し、前記符号データに基づきＤＣネストから選択生成した１又は２以上の基底ベクトルを合成してターゲットブロックの画像データを再生する画像復号装置において、

各ＤＣネスト画素の下位ｎ（$n=\log_2 B$）ビットが０にされているＤＣネストを記憶するメモリを備えることを特徴とする画像復号装置。

【請求項１６】　請求項１乃至１３の何れか１つに記載の処理をコンピュータに実行させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体に関し、更に詳しくはハイブリッドベクトル量子化（ＨＶＱ：Hybrid Vector Quantization）方式による画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体に関する。

【０００２】今日、静止画圧縮の国際標準であるＪＰＥＧ（Joint Photographic Expert Group）方式では、８×８の画素ブロックを２次元ＤＣＴによりＤＣ値及び基本～６３倍周波数の各係数値に変換すると共に、自然画の周波数成分が低周波領域に集中していることを利用して画品質が低下しない範囲内で各係数値を異なる量子化幅で量子化し、情報量の削減を行ってからハフマン符号化を行っている。

【０００３】これに対してＨＶＱ方式は、ＪＰＥＧと同様に平均値分離型ブロック符号化の一種であるが、ベクトル量子化と直交変換符号化の中間方式である適応的直交変換（ＡＯＴ：Adaptive Orthogonal Transform）をその圧縮原理としている。ここで、ＡＯＴはベクトル量子化のコードブックに相当する基底の巣（ネスト）から必要最少数の非直交基底系を選択し、対象ブロックを所望の許容誤差Ｚ以内に近似する方式である。ＨＶＱ方式では復号演算を整数型で行えるため、復号が高速である。またＪＰＥＧに特有なモスキート及びブロックノイズ、ＧＩＦに特有な擬似輪郭が発生しないため自然画像，人工画像（アニメーション画像，ＣＧ画像）を高画質で高圧縮できる。本発明はこのようなＨＶＱ方式における画質の更なる改善及び符号化演算の高速化に関する。

【０００４】

【従来の技術】本件出願人は画像の自己相似性を利用したＨＶＱ方式による画像符号／復号方法を既に提案している（特願平１０－１８９２３９）。以下その内容を説明する。なお、本明細書を通して記号〈ａ〉はベクトルａ又はブロックａ、記号‖ａ‖はベクトルａの大きさ（ノルム）、記号〈ａ・ｂ〉はベクトルａ，ｂの内積を表す。また図や［数］中のベクトルやブロックを太文字で表す。

【０００５】図１３は従来の画像符号装置（エンコーダ）のブロック図で、図において、１１は原画像データを記憶する原画像メモリ、１２は原画像データの各画素ブロック（４×４画素）につきブロック平均（ＤＣ）値を求めるＤＣ値生成部、１３は各ＤＣ値につき差分予測符号化を行う差分ＰＣＭ符号部（ＤＰＣＭ）、１４は差分ＰＣＭ符号から各ＤＣ値を復号する逆ＤＰＣＭ符号部（ＩＤＰＣＭ）、１５は復号ＤＣ画像を記憶するＤＣ画像メモリ、１６はＤＣ画像の一部から所定サイズのＤＣネストを切り出すＤＣネスト生成部、１７はＤＣネストを記憶するＤＣネストメモリである。

【０００６】更に、１８は符号対象であるターゲット画像ブロック〈R_j〉から対応する復号ＤＣ値DC_Jを分離する減算器、１９はＤＣ分離された残差ベクトル〈d_j〉を記憶する残差ベクトルバッファ、２０はＤＣネストからダウンサンプルされた４×４画素の基底候補ブロック〈U_i〉を記憶する候補ブロックバッファ、２１は基底候補ブロック〈U_i〉のブロック平均値a_iを求める平均器、２２は基底候補ブロック〈U_i〉からブロック平均値a_iを分離する減算器、２３は平均値分離された基底候補ベクトル〈u_i〉を記憶する候補ベクトルバッファ、２４は、残差ベクトルの二乗ノルム$\|d_j\|^2$が許容誤差Ｚを超える場合に、ＤＣネストを探索して残差ベクトル〈d_j〉を許容誤差Ｚ以内に近似するための直交基底系α_k〈u_k'〉（$k=1\sim m$）を生成する適応的直交変換処理部（ＡＯＴ）、２５は生成された直交基底系α_k〈u_k'〉（$k=1\sim m$）につき、各対応する非直交基底ベクトル〈u_k〉（$k=1\sim m$）に掛けて等価な非直交基底系β_k〈u_k〉（$k=1\sim m$）を生成するための展開係数βkを求める係数変換部、２６は上記ＤＣ値のＤＰＣＭ符号や非直交基底系β_k〈u_k〉等の情報を更に圧縮符号化するためのハフマン，ランレングス，固定長符号等による符号部である。

【０００７】ＤＣ値生成部１２は４×４画素のブロック平均値を求め、小数点以下を四捨五入（又は切り捨て等）する。ＤＰＣＭ１３は、図示しないが、Ｊ行，Ｉ列

のＤＣ値を$DC_{J,I}$とする時に、該$DC_{J,I}$の予測値$DC_{J,I}'$を例えば$DC_{J,I}'=(DC_{J,I-1}+DC_{J-1,I})/2$により求め、その予測誤差$\Delta DC_{J,I}=DC_{J,I}-DC_{J,I}'$を量子化係数Ｑ（Ｚ）により線形量子化して出力する。この量子化係数Ｑ（Ｚ）は許容誤差Ｚと対応付けられており、許容誤差Ｚに応じて１～８の範囲で変化する。

【０００８】ＤＣネスト生成部１６はＤＣ画像から例えば縦３９×横７１の領域をそのまま切り出（コピー）してＤＣネストとする。ＤＣネストはコードブックとして使用されるため、交流成分を多く含むものが望ましい。そこで、複数の候補領域につき、各領域内で隣り合うＤＣ値の差分をとってこれらの絶対値等の総和を求め、総和が最大となるような領域を切り出してＤＣネストとする。

【０００９】また基底候補ブロック$\langle U_i \rangle$のダウンサンプルは、縦横１ＤＣ値毎に頂点（ｐｘ，ｐｙ）∈［０，６３］×［０，３１］を設定し、かつそのサブサンプル間隔は（ｓｘ，ｓｙ）∈｛（１，１），（１，２），（２，１），（２，２）｝の計４種類とする。従って、トータルではＮ（＝８１９２）個の基底候補ブロック$\langle U_i \rangle$が存在し、これらはＡＯＴ２４からのインデクスカウンタｉで参照される。以下、従来の適応的直交変換処理部２４の動作を説明する。

【００１０】図１４は従来の適応的直交変換処理のフローチャート、図１５で該処理のイメージ図である。図１４において、残差ベクトルの二乗ノルム$\|\langle d_j \rangle\|^2>Z$であるとこの処理に入力する。ステップＳ１２１ではレジスタＥに残差ベクトルの二乗ノルム$\|\langle d_j \rangle\|^2$をセットする。また基底数カウンタｋ＝１に初期化する。ステップＳ１２２では最小値保持レジスタＥ’に大きな値（例えば１０００００）をセットする。ステップＳ１２３では基底候補ブロック$\langle U_i \rangle$のインデクスカウンタｉ＝０に初期化する。これはＤＣネストの開始アドレス（ｐｘ，ｐｙ）＝（０，０），サブサンプル間隔（ｓｘ，ｓｙ）＝（１，１）に対応する。

【００１１】ステップＳ１２４では基底候補ブロック$\langle U_i \rangle$からそのブロック平均値ａｉを分離して基底候補ベクトル$\langle u_i \rangle$を生成する。この演算は整数精度で行われるため、ブロック平均値a_iに小数点以下の値が発生した場合はこれを四捨五入（又は切り捨て等）する。ステップＳ１２５では必要（ｋ＞１）なら基底候補ベクトル$\langle u_i \rangle$をそれ以前の直交基底ベクトル$\langle u_k' \rangle$に直交化する。

【００１２】図１５（Ａ），（Ｂ）に直交化処理のイメージ図を示す。図１５（Ａ）において、まず第１基底候補ベクトル$\langle u_1 \rangle$はそのままで第１基底ベクトル$\langle u_1' \rangle$となり得る。次に第２基底候補ベクトル$\langle u_2 \rangle$は以下の方法により第１基底ベクトル$\langle u_1' \rangle$に直交化される。即ち、第２基底候補ベクトル$\langle u_2 \rangle$の第１基底ベクトル$\langle u_1' \rangle$への射影は（１）式の関係で得られる．

【００１３】

【数１】

$$\|u_2\|\cos\theta=\frac{\langle u_1'\bullet u_2\rangle}{\|u_1'\|} \qquad \because \quad \langle u_1'\bullet u_2\rangle=\|u_1'\|\|u_2\|\cos\theta \qquad (1)$$

【００１４】従って、第２直交ベクトル$\langle u_2' \rangle$は第２基底候補ベクトル$\langle u_2 \rangle$から前記射影分のベクトルを引くことで得られる。

【００１５】

【数２】

$$u_2'=u_2-\frac{\langle u_1'\bullet u_2\rangle}{\|u_1'\|}\frac{u_1'}{\|u_1'\|} \qquad (2)$$

【００１６】図１５（Ｂ）において、次に第３基底候補ベクトル$\langle u_3 \rangle$を第１，第２の基底ベクトル$\langle u_1' \rangle$，$\langle u_2' \rangle$に直交化する。この図は３次元的に描かれている。まず第３基底候補ベクトル$\langle u_3 \rangle$を第１基底ベクトル$\langle u_1' \rangle$に直交化すると上記同様にして中間の直交ベクトル$\langle u_3'' \rangle$が得られる．

【００１７】

【数３】

$$u_3''=u_3-\frac{\langle u_1'\bullet u_3\rangle}{\|u_1'\|^2}u_1' \qquad (3)$$

【００１８】更にこの中間直交ベクトル$\langle u_3'' \rangle$を第２基底ベクトル$\langle u_2' \rangle$に直交化すると第３基底ベクトル$\langle u_3' \rangle$が得られる。

【００１９】

【数４】

$$
\begin{aligned}
\mathbf{u}_3' &= \mathbf{u}_3'' - \frac{\left\langle \mathbf{u}_2' \bullet \mathbf{u}_3'' \right\rangle}{\left\| \mathbf{u}_2' \right\|^2} \mathbf{u}_2' \\
&= \left(\mathbf{u}_3 - \frac{\left\langle \mathbf{u}_1' \bullet \mathbf{u}_3 \right\rangle}{\left\| \mathbf{u}_1' \right\|^2} \mathbf{u}_1' \right) - \frac{\left\langle \left(\mathbf{u}_3 - \frac{\left\langle \mathbf{u}_1' \bullet \mathbf{u}_3 \right\rangle}{\left\| \mathbf{u}_1' \right\|^2} \mathbf{u}_1' \right) \bullet \mathbf{u}_2' \right\rangle}{\left\| \mathbf{u}_2' \right\|^2} \mathbf{u}_2' \\
&= \mathbf{u}_3 - \frac{\left\langle \mathbf{u}_1' \bullet \mathbf{u}_3 \right\rangle}{\left\| \mathbf{u}_1' \right\|^2} \mathbf{u}_1' - \frac{\left\langle \mathbf{u}_2' \bullet \mathbf{u}_3 \right\rangle}{\left\| \mathbf{u}_2' \right\|^2} \mathbf{u}_2' \qquad (4)
\end{aligned}
$$

【００２０】図１３に戻り、ステップＳ１２６では得られた直交ベクトル〈u_i'〉を使用し、残差ベクトル〈d_k〉（但し、最初は〈d_j〉）との距離を最小とする様なスカラー係数α_iを求める。

【００２１】図１５（Ｃ）にその処理イメージを示す。図において、ある時点の残差ベクトルを〈d_k〉とする時に、これを直交ベクトル〈u_i'〉で近似した後の残差ベクトルの二乗ノルムｅｉ＝‖〈d_k〉－α_i〈ｕｉ'〉‖^{2}が最小となるのは、図より明らかなように、直交ベクトル〈ｕｉ'〉にスカラー係数α_iを掛けたものと、残差ベクトル｛〈d_k〉－α_i〈u_i'〉｝とが直交する時（内積＝０）である。従って、スカラー係数α_iは（５）式の関係により求まる。

【００２２】

【数５】

$$
\left\langle \alpha_i \mathbf{u}_i' \bullet \left(\mathbf{d}_k - \alpha_i \mathbf{u}_i' \right) \right\rangle = 0 \qquad \alpha_i \left\langle \mathbf{u}_i' \bullet \mathbf{d}_k \right\rangle - \alpha_i^2 \left\langle \mathbf{u}_i' \bullet \mathbf{u}_i' \right\rangle = 0 \qquad (5\text{-}1)
$$

$$
\alpha_i = \frac{\left\langle \mathbf{d}_k \bullet \mathbf{u}_i' \right\rangle}{\left\| \mathbf{u}_i' \right\|^2} \qquad (5\text{-}2)
$$

【００２３】なお、図には残差ベクトル〈d_k〉（但し、ｋ＝０）を他の第１基底候補ベクトル〈u_j'〉で近似した場合が描かれている。第１基底候補ベクトル〈u_j'〉は任意方向をとり得るから、図示のようなイメージとなる。

【００２４】図１４に戻り、ステップＳ１２７では残差ベクトル〈d_k〉を基底候補ベクトルα_i〈u_i'〉で近似した後の誤差ベクトルの二乗ノルムe_iを求める。この演算は（６）式により得られる。

【００２５】

【数６】

$$
\begin{aligned}
e_i &= \left\| \mathbf{d}_k - \alpha_i \mathbf{u}_i' \right\|^2 \\
&= \left\| \mathbf{d}_k \right\|^2 - 2\alpha_i \left\langle \mathbf{d}_k \bullet \mathbf{u}_i' \right\rangle + \alpha_i^2 \left\| \mathbf{u}_i' \right\|^2 \\
&= \left\| \mathbf{d}_k \right\|^2 - 2\frac{\left\langle \mathbf{d}_k \bullet \mathbf{u}_i' \right\rangle^2}{\left\| \mathbf{u}_i' \right\|^2} + \frac{\left\langle \mathbf{d}_k \bullet \mathbf{u}_i' \right\rangle^2}{\left\| \mathbf{u}_i' \right\|^4} \left\| \mathbf{u}_i' \right\|^2 \\
&= \left\| \mathbf{d}_k \right\|^2 - \frac{\left\langle \mathbf{d}_k \bullet \mathbf{u}_i' \right\rangle^2}{\left\| \mathbf{u}_i' \right\|^2} \\
&= E - \frac{\left\langle \mathbf{d}_k \bullet \mathbf{u}_i' \right\rangle^2}{\left\| \mathbf{u}_i' \right\|^2} \qquad (6)
\end{aligned}
$$

【００２６】ステップＳ１２８ではe_i＜Ｅ'か否かを判別する。e_i＜Ｅ'の場合はステップＳ１２９でＥ'の内容をe_iで更新する。またその時のα_i，〈u_i'〉，〈u_i〉等に係る情報を配列［α_k］，［u_k'］，［u_k］に保持する。またe_i＜Ｅ'でない場合は上記ステップＳ１２９の処理をスキップする。

【００２７】ステップＳ１３０ではカウンタｉに＋１し、更にステップＳ１３１ではｉ≧Ｎ（＝８１９２）か否かを判別する。ｉ≧Ｎでない場合はステップＳ１２４に戻り、次の基底候補ベクトル〈u_i〉につき上記同様の処理を行う。以下同様にして進み、やがて、ステップＳ１３１の判別でｉ≧Ｎになるとこの段階における全基底候補ベクトル〈u_i〉が試されたことになる。この時、レジスタＥ'は最小の二乗ノルムe_iを保持している。

【００２８】ステップＳ１３２ではＥ'≦Ｚか否かを判別し、Ｅ'≦Ｚでない場合はステップＳ１３３でＥ＝Ｅ'とする。即ち、残差ベクトルの二乗ノルムを更新する。ステップＳ１３４ではｋに＋１し、ステップＳ１２２に戻る。またＥ'≦Ｚの場合はこの処理を抜ける。こ

うして、最初の残差ベクトル〈d_j〉との差を許容誤差Z以下に近似するための直交基底系α_k〈u_k'〉（k＝1～m）が得られる。

【００２９】

【発明が解決しようとする課題】しかし、上記従来方式では基底候補ブロック〈U_i〉のブロック平均値a_iにつきその小数点以下を四捨五入（又は切り捨て等）していたため、画質の改善が頭打ちとなる不都合があった。これを図１６に従って説明する。

【００３０】図１６（ａ）は基底候補ブロック〈U_i〉のある行の画素値を列（ｘ）方向に見た場合を示している。実際は１６画素分のブロック平均値であるが、ここでは説明の簡単のため４画素で説明する。図１６（ａ）において、各画素値は「５，２，４，３」からなりそのブロック平均値a_i＝３．５である。今、例えばこの小数点以下を切り捨てるとすると、図１６（ｂ）に示す如く、基底候補ベクトル〈u_i〉のブロック平均値a_i＝０．５となる。図１６（ｃ）において、復号ブロックのＤＣ値DC_Jに基底ベクトルβ_k〈u_k〉を加算すると、復号画像のターゲットブロック〈R_j〉にはＤＣ成分（a_i＝０．５）が重畳されてしまう．しかも、基底数が複数の場合は、このようなＤＣ成分は$0<a_i<1$の範囲の様々な値でDC_Jに重畳される結果、復号画像ではブロック毎に一種の雑音が重畳された形となり、このため画質の改善が図れなかった。以上のことは小数点以下を四捨五入又は切り上げする場合も同様である。

【００３１】また、従来のＡＯＴ処理では各基底候補ベクトル〈u_i〉を一々前の基底ベクトル〈u_k'〉に直交化していたため、ＡＯＴ処理に多大の演算と時間を要していた。

【００３２】本発明は上記従来技術の問題点に鑑み成されたもので、その目的とする所は、より高画質かつ高速の符号／復号が得られる画像符号／復号方法及びその装置並びにそのプログラムを記録した記録媒体を提供することにある。

【００３３】

【課題を解決するための手段】上記の課題は例えば図１の構成により解決される。即ち、本発明（１）の画像符号方法は、画像データをＢ画素毎にブロック分割して各ブロック平均値からなるＤＣ画像を生成し、その一部をＤＣネストとすると共に、符号対象の画素ブロック〈R_j〉からそのＤＣ値DC_Jを分離した残差ベクトル〈d_j〉の大きさが許容値Ｚを超える場合は、該残差ベクトル〈d_j〉を近似するための１又は２以上の直交基底（α_k〈v_k〉等）をＤＣネストを使用した適応的直交変換（ＡＯＴ）により求める画像符号方法において、ＤＣネストから基底候補ブロック〈U_i〉をダウンサンプルしてそのブロック平均値a_iを求める際の各サンプルＤＣ画素の下位ｎ（$n=\log_2 B$）ビットが０にされているものである。従って、ブロック平均値a_iに小数点以下の端数は生ぜず、整数精度のブロック平均値a_iが高速に得られる。

【００３４】好ましくは本発明（２）においては、上記本発明（１）において、ＤＣ画像からＤＣネストを生成する際に各ＤＣ画素の下位ｎビットを０に（マスク）する。従って、１回の処理でその下位ｎビットを０にされたＤＣネストが効率よく得られる。

【００３５】また好ましくは本発明（３）においては、上記本発明（１）又は（２）において、下位ｎビットが０にされている基底候補ブロック〈U_i〉からそのブロック平均値a_iを分離して残差ベクトル〈d_j〉を近似するための基底候補ベクトル〈u_i〉を生成する。

【００３６】本発明（３）によれば、このような基底候補ベクトル〈u_i〉はその全要素の和（ブロック平均値）が常に０であり、ＤＣ成分が完全に分離されている。従って、復号側でこのような基底ベクトル〈u_k〉を幾つ重ねても不要なＤＣ成分（雑音）は生じない。そして、これにより本ＨＶＱ方式の画質が大幅に改善された。

【００３７】また好ましくは本発明（４）においては、上記本発明（３）において、基底候補ベクトル〈u_i〉の任意要素（例えばu_{16}）を残りの要素の一次結合で置き換えると共に、該基底候補ベクトル〈u_i〉と任意他のベクトル〈ｗ〉との内積を、

$$\langle w\cdot u_i\rangle=(w_1-w_{16})\,u_1+(w_2-w_{16})\,u_2+,\ \cdots,\ +(w_{15}-w_{16})\,u_{15}$$

の積和演算により求める。

【００３８】本発明（４）においては、上記基底候補ベクトル〈u_i〉の全要素の和が常に０であることにより、その任意要素（例えばu_{16}）を残りの要素の一次結合で表せる。従って、任意他のベクトル〈ｗ〉との内積演算〈$w\cdot u_i$〉は上式のような積和演算に展開でき、こうして面倒な積和演算の回数を１回分省略できる。ＨＶＱ方式による画像符号処理ではベクトルの内積演算が大量に行われるため、各１回の省略は全体としての符号処理の高速化に大きく貢献する。

【００３９】また好ましくは本発明（５）においては、上記本発明（３）又は（４）において、残差ベクトル〈ｄ〉、基底候補ベクトル〈u_i〉とするときに第１基底の探索は、

$$h_i=\langle d\cdot u_i\rangle^2/\|u_i\|^2$$

を最大とするものを条件に探索する。

【００４０】本発明（５）によれば、図１４の従来のステップＳ１２６，Ｓ１２７で行っていたような残差ベクトル〈ｄ〉との差の二乗ノルム$\|\langle d\rangle-\alpha_i\langle u_i\rangle\|^2$を最小とするような条件を上記簡単な演算及び条件で探索できる。従って、ＡＯＴ処理を高速化できる。

【００４１】また好ましくは本発明（６）においては、上記本発明（３）又は（４）において、残差ベクトル〈ｄ〉、第１基底に対応する基底候補ベクトル

$\langle u_1\rangle$、第２基底を探索する基底候補ベクトル$\langle u_i\rangle$とするときに第２基底の探索は、

$$h_i=\{\langle d\cdot u_i\rangle-(\langle d\cdot u_1\rangle\langle u_1\cdot u_i\rangle)/\|u_1\|^2\}^2/\{\|u_i\|^2-(\langle u_1\cdot u_i\rangle)/\|u_1\|)^2\}$$

を最大とするものを条件に探索する。

【００４２】本発明（６）によれば、上記本発明（５）の効果に加え、上式分子の$\langle d\cdot u_1\rangle$，$\|u_1\|$及び分母の$\|u_i\|^2$，$\|u_1\|$については既に第１基底探索で行った演算結果を利用できるため、ＡＯＴ処理を更に効率化、高速化できる。

【００４３】また好ましくは本発明（７）においては、上記本発明（３）又は（４）において、残差ベクトル$\langle d\rangle$、第１正規基底ベクトル$\langle v_1\rangle$、第２正規直交基底ベクトル$\langle v_2\rangle$、第３基底を探索する基底候補ベクトル$\langle u_i\rangle$とするときに第３基底の探索は、

$$h_i=(\langle d\cdot u_i\rangle-\langle d\cdot v_1\rangle\langle v_1\cdot u_i\rangle-\langle d\cdot v_2\rangle\langle v_2\cdot u_i\rangle)^2/\{\|u_i\|^2-\langle v_1\cdot u_i\rangle^2-\langle v_2\cdot u_i\rangle^2\}$$

を最大とするものを条件に探索する。

【００４４】本発明（７）によれば、上記本発明（５），（６）の効果に加え、上式分子の$(\langle d\cdot u_i\rangle-\langle d\cdot v_1\rangle\langle v_1\cdot u_i\rangle)$及び分母の$(\|u_i\|^2-\langle v_1\cdot u_i\rangle^2)$については既に第１，第２の基底探索で行った演算結果を利用できるため、こうしてＡＯＴ処理を更に効率化、高速化できる。

【００４５】また好ましくは本発明（８）においては、上記本発明（６）又は（７）において、探索条件にマッチした基底候補ベクトル$\langle u_i\rangle$をそれ以前の１又は２以上の正規直交基底に正規直交化する。

【００４６】即ち、このような正規直交化は各段階の探索終了により基底に採用された各基底候補ベクトル$\langle u_i\rangle$につき夫々１回だけ行えばよく、こうしてＡＯＴ処理を更に効率化、高速化できる。

【００４７】また本発明（９）の画像符号方法は、上記前提となる画像符号方法において、求められた基底系を$\beta_k\langle u_k\rangle$（$k=1\sim m$）とするときに、各スカラー展開係数$\beta_1\sim\beta_m$のノルムをその大きさ順に並べ替え、０を含む隣接ノルム間の各差分を求め、得られた各差分につきハフマン符号を適用するものである。

【００４８】一般にスカラー展開係数$\beta_1\sim\beta_m$のノルムは様々な値をとり得るが、これらを大きさ順に並べて０を含む隣接ノルム間の各差分をとると、各差分の大きさは互いに近似（又は同一）となる場合が少なくない。そこで、これらの差分値にハフマン符号を適用することで更なる符号圧縮が可能となる。

【００４９】また本発明（１０）の画像符号方法は、上記前提となる画像符号方法において、求めた基底数が所定以上の場合は基底系の符号化に代えて、符号対象ブロックの画像データ$\langle R_j\rangle$そのものを符号化するものである。従って、復号画質の改善が図れる。また実際上このような状況は極めて少ないので符号圧縮率に与える影響は極めて少ない。

【００５０】また上記の課題は例えば図１０の構成により解決される。即ち，本発明（１１）の画像復号方法は、ＨＶＱ方式に係る符号データからＢ画素毎の各ブロック平均値に相当するＤＣ画像を再生し、その一部をＤＣネストとすると共に、ターゲットブロックのＤＣ値DC_Jに対し、前記符号データに基づきＤＣネストから選択生成した１又は２以上の基底ベクトル$\beta_k\langle u_k\rangle$を合成してターゲットブロックの画像データ$\langle R_j\rangle$をを再生する画像復号方法において、ＤＣネストから選択ブロック$\langle U_k\rangle$をダウンサンプルしてそのブロック平均値を求める際の各サンプルＤＣ画素の下位ｎ（$n=\log_2B$）ビットが０にされているものである。従って、ブロック平均値に小数点以下の端数は生ぜず、整数精度のブロック平均値が高速に得られる。

【００５１】また本発明（１２）の画像復号方法は、上記前提となる画像復号方法において、復号された基底系が$\beta_k\langle u_k\rangle$（$k=1\sim m$）に係る情報であるときに、ＤＣネストから読み出した各選択ブロック$\langle U_k\rangle$につき各ＤＣ画素の下位ｎ（$n=\log_2B$）ビットが０にされていると共に、まず$\beta_k\langle U_k\rangle$（$k=1\sim m$）の積和演算を行い、その演算結果をブロック画素数Ｂで除算するものである。

【００５２】本発明（１２）においては、各選択ブロック$\langle U_k\rangle$の下位ｎビットが０にされていることにより、これらを予め累積加算しても、その加算結果はブロックサイズＢ（例えば１６）の整数倍となる。なお、展開係数β_kは整数精度とする。従って、最後にこの累積加算結果をブロック画素数Ｂで除算すれば１回の除算でブロック平均値A_jが効率よく求まる。従って、基底ベクトル$\beta_k\langle u_k\rangle$（$k=1\sim m$）を重ね合わせる演算を効率よく行える。

【００５３】好ましくは本発明（１３）においては、上記本発明（１１）又は（１２）において、ＤＣ画像からＤＣネストを生成する際に各ＤＣ画素の下位ｎビットを０にする。従って、処理効率が良い。

【００５４】また本発明（１４）の画像符号装置は、画像データをＢ画素毎にブロック分割して各ブロック平均値からなるＤＣ画像を生成し、その一部をＤＣネストとすると共に、符号対象の画素ブロック$\langle R_j\rangle$からそのＤＣ値DC_Jを分離した残差ベクトル$\langle d_j\rangle$の大きさが許容値Ｚを超える場合は、該残差ベクトル$\langle d_j\rangle$を近似するための１又は２以上の直交基底（$\alpha_k\langle v_k\rangle$等）をＤＣネストを使用した適応的直交変換（ＡＯＴ）により求める画像符号装置において、各ＤＣネスト画素の下位ｎ（$n=\log_2B$）ビットが０にされているＤＣネストを記憶するメモリ１７を備えるものである。

【００５５】また本発明（１５）の画像復号装置は、Ｈ

ＶＱ方式に係る符号データからＢ画素毎の各ブロック平均値に相当するＤＣ画像を再生し、その一部をＤＣネストとすると共に、ターゲットブロックのＤＣ値DC_Jに対し、前記符号データに基づきＤＣネストから選択生成した１又は２以上の基底ベクトルβ_k〈u_k〉を合成してターゲットブロックの画像データ〈R_j〉をを再生する画像復号装置において、各ＤＣネスト画素の下位ｎ（ｎ$=\log_2 B$）ビットが０にされているＤＣネストを記憶するメモリ４９を備えるものである。

【００５６】また本発明（１６）の記録媒体は、上記本発明（１）１乃至（１３）の何れか１つに記載の処理をコンピュータに実行させるためのプログラムを記録したコンピュータ読取り可能な記録媒体である。

【００５７】

【発明の実施の形態】以下、添付図面に従って本発明に好適なる実施の形態を詳細に説明する。なお、全図を通して同一符号は同一又は相当部分を示すものとする。

【００５８】図２は実施の形態による画像符号装置のブロック図で、図において、３１は復号ＤＣ画像から本発明によるＤＣネストを生成するＤＣネスト生成部、１７は生成されたＤＣネストを記憶するＤＣネストメモリ、３２はＡＯＴ処理を効率よくかつ高速に行う適応的直交変換処理部（ＡＯＴ）、３３は係数変換部、３４は展開係数β_kの更に高圧縮を可能とする符号部である。その他の構成については上記図１３で述べたものと同様でよい。なお、上記各部の特徴は以下の動作説明によって明らかとなる。

【００５９】図３は実施の形態による画像符号（メイン）処理のフローチャートである。ステップＳ１では原画像メモリ１１に原画像データを読み込む。例えばＲＧＢ系の対象画像をＹＵＶ系に変換して読み込む。Ｙは輝度データ、Ｕ，Ｖは色差データに相当し、Ｕ，Ｖは横２画素の輝度平均を用いてダウンサンプリングされる。一例の輝度データＹは縦９６０×横１２８０画素からなり、画素毎に例えば８ビットが割り付けられている。なお、以下は輝度データＹの処理を中心に述べるが、Ｕ，Ｖについても同様に処理できる。

【００６０】ステップＳ２では全画像データにつき４×４画素毎のブロック平均（ＤＣ）値を求める。このとき小数点以下は例えば四捨五入される。ステップＳ３では全ＤＣ値を公知の２次元ＤＰＣＭ法等により符号化して出力する。ステップＳ４では全ＤＰＣＭ出力をＩＤＰＣＭ復号してＤＣ画像を再生し、ＤＣ画像メモリ１５に格納する。これは符号側／復号側のＡＯＴ処理条件を同一にするためである。ステップＳ５ではＤＣネスト生成部３１がＤＣ画像からＤＣネストを生成し、ＤＣネストメモリ１７に格納する。なお、ＤＣネストを切り出す領域の選択等は従来と同様でよい。

【００６１】図７にＤＣネストの生成イメージを示す。図７（ａ）において、本実施の形態ではＤＣ画像メモリ１５から切り出した各ＤＣ画素DC_Jの下位４ビットをマスク（＝０）してこれをＤＣネストメモリ１７のネスト画素N_Jに記憶する。下位４ビットは$2^4=B$（Ｂ＝ブロックサイズ１６）又は$4=\log_2 B$の関係にある。下位４ビットをマスクした結果、基底候補ブロック〈U_i〉の総和は常に１６の整数倍となり、よってこれを１／１６したブロック平均値a_iは常に整数となる。従って、基底候補ブロック〈U_i〉からブロック平均値a_iを分離した基底候補ベクトル〈u_i〉のブロック平均値は常に０となる。

【００６２】図７（ａ），（ｂ）に具体的な数値例をグラフで示す。但し、ここでは説明の簡単のため４画素分の平均をとっている。図７（ｃ）において、復号ブロック〈R_j〉のＤＣ値DC_Jに複数の基底ベクトルβ_k〈u_k〉を累積加算しても、各基底ベクトルβ_k〈u_k〉のブロック平均値は常に０であるため、従来のような雑音は重畳されない。これにより画質の大幅な改善が図れた。

【００６３】図８（ａ）に図７の数値例を表で示す。ＤＣ画素Ａ～Ｄの合計ＳＵＭ＝２５１であり、その平均値ＡＶ＝２５１／４＝６２．７５（非整数）である。これらのＤＣ画素Ａ～Ｄをネスト画素Ａ～Ｄに転送する際に下位４ビットをマスクする。これによりネスト画素Ａ～Ｄの合計ＳＵＭ＝２２４となり、その平均値ＡＶ＝２２４／４＝５６（整数）となる。更に、ネスト画素Ａ～Ｄからその平均値ＡＶ＝５６を分離した基底候補ベクトル〈u_i〉の各要素ａ～ｄは「２４，－２４，８，－８」となり、これらの総和ｓｕｍ＝０（完全平均値分離）となっている。

【００６４】図８（ｂ）は図８（ａ）と同じ数値例を示している。但し、ＤＣ画素Ａ～Ｄをそのままネスト画素Ａ～Ｄにコピーし、ネスト画素Ａ～Ｄの総和ＳＵＭから下位４ビットをマスク（＝０）する点で異なっている。この方法でも総和ＳＵＭは１６の倍数になるから、ブロック平均値ＡＶ＝６０（整数）となる。しかしこの方法によると、ネスト画素Ａ～Ｄからその平均値ＡＶ＝６０を分離した基底候補ベクトル〈u_i〉の各要素ａ～ｄは「３３，－２５，１３，－１０」となり、必ずしもその総和ｓｕｍ＝０（完全平均値分離）とはならない。

【００６５】なお、図８（ｂ）に示す如くＤＣ画像の一部をそのままＤＣネストにコピーしておき、該ＤＣネストから基底候補ブロック〈U_i〉をダウンサンプルする時に各画素から下位４ビットをマスク（＝０）しても良い。

【００６６】図３に戻り、ステップＳ６では原画像メモリ１１及びＤＣ画像メモリ１５に対する各インデクスカウンタｊ，Ｊを共に０に初期化する。但し、ｊは符号対象のターゲットブロック〈R_j〉のインデクスカウンタ、ＪはＤＣ画素のインデクスカウンタを夫々表す。ステップＳ７ではターゲットブロック〈R_j〉から対応す

る復号DC値DC$_J$を分離して残差ベクトル〈d$_J$〉を求める。ステップS８では残差ベクトルの二乗ノルム‖d$_J$‖^{2}が許容誤差Ｚより大きいか否かを判別する。‖d$_J$‖^2＞Ｚでない場合はステップS１７で基底数「０」を符号出力する。この場合のターゲットブロック〈R$_J$〉は後述の交流成分予測法により復号される。また‖d$_J$‖^2＞Ｚの場合はステップS９で後述の適応的直交変換処理を行う。

【００６７】ステップS１０では適応的直交変換で生成された基底数ｋ＞４か否かを判別する。因みに、実測ではほとんどの場合にｋ＝１～３程度の統計結果が得られている。そこで、ｋ＞４の場合はステップS１８で基底数「５」を符号出力し、かつターゲットブロック〈R$_J$〉の各画素値を符号出力する。またｋ＞４でない場合はステップS１１で後述の展開係数β_kへの変換を行う。ステップS１２では基底数「ｍ」，展開係数β_k及び非直交基底ベクトル〈u$_i$〉のインデクス情報ｉを夫々符号出力する。

【００６８】ステップS１３ではカウンタｊ，Ｊに夫々＋１する。但し、カウンタｊに対する＋１は１画素ブロック分の更新を意味する。ステップS１４ではｊ≧Ｍ（＝全画像ブロック数）か否かを判別する。ｊ≧Ｍでない場合はステップS７に戻り、次のターゲットブロック〈R$_J$〉につき上記同様の符号処理を行う。以下同様にして進み、やがて、ステップS１４の判別でｊ≧Ｍになると、ステップS１５ではハフマン等による符号化を行う。この符号化については後述する。こうして１画像分の符号処理を終了する。

【００６９】図４～図６は実施の形態による適応的直交変換処理のフローチャート（１）～（３）で、必要最少数の直交基底系α_k〈v$_k$〉（ｋ＝１～ｍ）を効率よくかつ高速に求め得る場合を示している。なお、以下の説明では上記ステップS７で求められた最初の残差ベクトル〈d$_J$〉を〈d〉で表し、その後に更新される残差ベクトルを〈d$_k$〉（ｋ＝１～ｍ）で表す。

【００７０】図４は第１基底の探索処理を示している。この処理の説明前に、該処理を高速に行うために行った計算上の工夫を説明する。即ち、通常なら第１基底は残差ベクトル〈d〉との差の二乗ノルムe$_i$を最小とする基底候補ベクトル〈u$_i$〉として求められるが、この関係式を更に展開すると（７）式が得られる。

【００７１】

【数７】

$$
\begin{aligned}
e_i &= \left\| \mathbf{d} - \frac{\langle \mathbf{d} \bullet \mathbf{u}_i \rangle}{\|\mathbf{u}_i\|^2} \mathbf{u}_i \right\|^2 \\
&= \|\mathbf{d}\|^2 - 2\frac{\langle \mathbf{d} \bullet \mathbf{u}_i \rangle^2}{\|\mathbf{u}_i\|^2} + \frac{\langle \mathbf{d} \bullet \mathbf{u}_i \rangle^2}{\|\mathbf{u}_i\|^4}\|\mathbf{u}_i\|^2 \\
&= \|\mathbf{d}\|^2 - \frac{\langle \mathbf{d} \bullet \mathbf{u}_i \rangle^2}{\|\mathbf{u}_i\|^2} \qquad \text{但し、} \quad \mathbf{u}_i = \mathbf{u}_i' \qquad (7)
\end{aligned}
$$

【００７２】ところで、（７）式右辺第１項の‖d‖^2＞０は基底候補によらず一定であるから、同右辺第２項を最大にする〈u$_i$〉が第１基底となり得る。そこで、この右辺第２項をh$_i$とおく。

【００７３】

【数８】

$$
h_i = \frac{\langle \mathbf{d} \bullet \mathbf{u}_i \rangle^2}{\|\mathbf{u}_i\|^2} \qquad (8)
$$

【００７４】以下は、h$_i$を最大とするような第１基底α_k〈v$_k$〉を探索・決定する処理である。ステップS２１では後述の内積演算〈d・u$_i$〉の前処理として〈d〉の第１６成分を残りの各成分の値から差し引いた１５次元ベクトル〈d'〉を求める。ステップS２２ではｉ＝０～（Ｎ－１）につきh$_i$分子の内積〈d'・u$_i$〉を求め、これらを配列［P$_i$］｛ｉ＝０～（Ｎ－１）｝に格納する。

【００７５】この内積演算を具体的に言うと、〈u$_i$〉は本来１６次元ベクトルであるが、本実施の形態ではそのブロック平均値（全要素の和）＝０により、その第１６成分u$_{16}$は残りの１５成分の一次結合で表せる。

【００７６】

【数９】

$$
\begin{aligned}
&\mathbf{u}_i = [u_1, u_2, u_3, \cdots, u_{16}] \\
&u_1 + u_2 + \cdots + u_{16} = 0 \\
&u_{16} = -(u_1 + u_2 + \cdots + u_{15}) \qquad (9)
\end{aligned}
$$

【００７７】従って、h$_i$分子の内積〈d・u$_i$〉をこれと等価な〈d'・u$_i$〉により求め、こうして積和演算を１回分（全ｉでは８１９２回分）省略できる。

【００７８】

【数１０】

$$
\begin{aligned}
\langle \mathbf{d} \bullet \mathbf{u}_i \rangle &= d_1 u_1 + d_2 u_2 + \cdots + d_{15} u_{15} - d_{16}(u_1 + u_2 + \cdots + u_{15}) \\
&= (d_1 - d_{16}) u_1 + (d_2 - d_{16}) u_2 + \cdots + (d_{15} - d_{16}) u_{15} \\
&= \langle \mathbf{d}' \bullet \mathbf{u}_i \rangle \qquad (10\text{-}1)
\end{aligned}
$$

$$
\mathbf{d}' = [(d_1 - d_{16}), (d_2 - d_{16}), \cdots, (d_{15} - d_{16})] \qquad (10\text{-}2)
$$

【００７９】ステップＳ２３ではｉ＝０〜（Ｎ−１）につきh_i分母の二乗ノルム‖u_i‖^{2}を求め、これらを配列［L_i］｛ｉ＝０〜（Ｎ−１）｝に格納する。

【００８０】

【数１１】

$$
\|\mathbf{u}_i\|^2 = u_1^2 + u_2^2 + \cdots + u_{16}^2 \qquad (11)
$$

【００８１】配列［L_i］は一度求めておけば後に結果を利用できる。ステップＳ２４ではh_iの最大値を保持するレジスタＥ＝０，基底候補ベクトル〈u_i〉のインデクスカウンタｉ＝０，基底数カウンタｋ＝１に夫々初期化する。

【００８２】ステップＳ２５ではh_i＝P_i^2／L_iを求める。ステップＳ２６ではh_i＞Ｅか否かを判別する。h_i＞Ｅの場合はステップＳ２７でＥをh_iで更新し、かつその時のｉを配列［I_k］（ｋ＝１）に保持する。またh_i＞Ｅでない場合は上記ステップＳ２７の処理をスキップする。

【００８３】ステップＳ２８ではｉに＋１し、更にステップＳ２９ではｉ≧Ｎ（全候補数）か否かを判別する。ｉ≧Ｎでない場合はステップＳ２５に戻り、次のh_iにつき上記同様の最大値探索処理を行う。以下、同様にして進み、やがてｉ≧Ｎになると全ネストブロックの探索終了である。この時、上記配列［I_k］にはh_iを最大とするような第１基底ベクトル〈u_1〉のインデクス値ｉが保持されている。

【００８４】ステップＳ３０では第１基底ベクトル〈u_1〉を正規化して正規化基底ベクトル〈v_1〉となし、これを配列［V_k］（ｋ＝１）に格納する。またスカラー係数α_1（〈ｄ〉の〈v_1〉への射影）を求め、これを配列［A_k］（ｋ＝１）に格納する。

【００８５】ステップＳ３１では残差ベクトル〈ｄ〉を第１基底で近似後の残差ベクトル〈d_1〉＝〈ｄ〉−α_1〈v_1〉により更新する。ステップＳ３２では新たな残差ベクトルの二乗ノルムｅ＝‖d_1‖^{2}を求め、更にステップＳ３３ではｅ≦Ｚか否かを判別する。ｅ≦Ｚの場合はこの段階でＡＯＴ処理を終了し、またｅ≦Ｚでない場合は次いで第２基底の探索処理を行う。

【００８６】図５は第２基底の探索処理を示している。この処理の説明前に該処理を効率よく行うために行った計算上の工夫を説明する。即ち、通常なら第２基底は残差ベクトル〈d_1〉との差の二乗ノルムe_iを最小とする直交ベクトル〈u_i'〉として求められるが、この関係式を更に展開すると（１２）式が得られる。

【００８７】

【数１２】

$$
\begin{aligned}
e_i &= \left\| \mathbf{d}_1 - \frac{\langle \mathbf{d}_1 \bullet \mathbf{u}_i' \rangle}{\|\mathbf{u}_i'\|^2} \mathbf{u}_i' \right\|^2 \\
&= \|\mathbf{d}_1\|^2 - 2\frac{\langle \mathbf{d}_1 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} + \frac{\langle \mathbf{d}_1 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^4}\|\mathbf{u}_i'\|^2 \\
&= \|\mathbf{d}_1\|^2 - \frac{\langle \mathbf{d}_1 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} \qquad (12)
\end{aligned}
$$

【００８８】ここで、直交ベクトル〈u_i'〉は第２基底の候補ベクトル＜u_i＞を第１正規化基底ベクトル＜v_1＞に直交化したものである。

【００８９】

【数１３】

$$
\mathbf{u}_i' = \mathbf{u}_i - \frac{\langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle}{\|\mathbf{v}_1\|^2}\mathbf{v}_1 = \mathbf{u}_i - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1 \qquad (13)
$$

【００９０】同様にして、上記（１２）式右辺第１項の‖d_1‖^2＞０は基底候補によらず一定であるから、同右辺第２項を最大にするような直交ベクトル〈u_i'〉が第２基底となり得る。この右辺第２項をh_iとおく。

【００９１】

【数１４】

$$h_i = \frac{\langle \mathbf{d}_1 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} \qquad (14)$$

【００９２】h_iは、このまま求めても良いが、上記図４の演算結果を効率よく利用するために（１４）式の分母を変形する。即ち、まずh_i分子の直交ベクトル〈u_i'〉を基底候補ベクトル〈u_i〉で表すと、h_i分子は（１５）式で表せる。

【００９３】

【数１５】

$$\begin{aligned}\langle \mathbf{d}_1 \bullet \mathbf{u}_i' \rangle^2 &= \langle \mathbf{d}_1 \bullet (\mathbf{u}_i - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1) \rangle^2 \\ &= (\langle \mathbf{d}_1 \bullet \mathbf{u}_i \rangle - \mathbf{d}_1 \bullet \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1)^2 \\ &= \langle \mathbf{d}_1 \bullet \mathbf{u}_i \rangle^2 \qquad \because \langle \mathbf{d}_1 \bullet \mathbf{v}_1 \rangle = 0 \end{aligned} \qquad (15)$$

【００９４】更に残差ベクトル〈d_1〉を最初の残差ベクトル〈d〉で表すと、h_i分子は（１６）式で表せる。

【００９５】

【数１６】

$$\begin{aligned}\langle \mathbf{d}_1 \bullet \mathbf{u}_i \rangle^2 &= \langle (\mathbf{d} - \langle \mathbf{d} \bullet \mathbf{v}_1 \rangle \mathbf{v}_1) \bullet \mathbf{u}_i \rangle^2 \\ &= (\langle \mathbf{d} \bullet \mathbf{u}_i \rangle - \langle \mathbf{d} \bullet \mathbf{v}_1 \rangle \langle \mathbf{v}_1 \bullet \mathbf{u}_i \rangle)^2 \\ &= \left(\langle \mathbf{d} \bullet \mathbf{u}_i \rangle - \frac{\langle \mathbf{d} \bullet \mathbf{u}_1 \rangle}{\|\mathbf{u}_1\|} \frac{\langle \mathbf{u}_1 \bullet \mathbf{u}_i \rangle}{\|\mathbf{u}_1\|} \right)^2 \end{aligned} \qquad (16)$$

【００９６】従って、h_i分子の演算には第１基底探索で得られた演算結果〈d・u_1〉，‖u_1‖を利用できる。また、同様にしてh_i分母を変形すると（１７）式で表せる。

【００９７】

【数１７】

$$\begin{aligned}\|\mathbf{u}_i'\|^2 &= \|\mathbf{u}_i - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1\|^2 \\ &= \|\mathbf{u}_i\|^2 - 2\langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle^2 + \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle^2 \|\mathbf{v}_1\|^2 \\ &= \|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle^2 \\ &= \|\mathbf{u}_i\|^2 - \left(\frac{\langle \mathbf{u}_i \bullet \mathbf{u}_1 \rangle}{\|\mathbf{u}_1\|} \right)^2 \end{aligned} \qquad (17)$$

【００９８】従って、h_i分母の演算には第１基底探索で得られた演算結果‖u_i‖^2，‖u_1‖を利用できる。以上を（１４）式のh_iに代入すると、h_iは（１８－１）式となり、最終的に（１８－２）式で表せる。

【００９９】

【数１８】

$$\begin{aligned} h_i &= \frac{\left(\langle \mathbf{d} \bullet \mathbf{u}_i \rangle - \frac{\langle \mathbf{d} \bullet \mathbf{u}_1 \rangle}{\|\mathbf{u}_1\|} \frac{\langle \mathbf{u}_1 \bullet \mathbf{u}_i \rangle}{\|\mathbf{u}_1\|} \right)^2}{\|\mathbf{u}_i\|^2 - \left(\frac{\langle \mathbf{u}_i \bullet \mathbf{u}_1 \rangle}{\|\mathbf{u}_1\|} \right)^2} && (18\text{-}1) \\ &= \frac{\left(P_i - \frac{P_k}{\sqrt{L_k}} \frac{\langle \mathbf{u}_k \bullet \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2}{L_i - \left(\frac{\langle \mathbf{u}_k \bullet \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2} && (18\text{-}2) \end{aligned}$$

【０１００】ここで、P_i＝〈d・u_i〉，L_i＝‖u_i‖^{2}は配列［P_i］，［L_i］の演算結果を夫々利用でき、かつP_k＝P_1＝〈d・u_1〉，√（L_k）＝√（L_1）＝‖u_1‖も前回の演算結果を利用できる。従って、今回新たに演算するのは〈u_k・u_i〉＝〈u_1・u_i〉の部分である。

【０１０１】以上を前提として、第２基底の探索は以下の演算処理を行う。即ち、ステップＳ４１ではｋ＝１によりP_1＝〈d・u_1〉，L_1＝‖u_1‖^{2}を保持する。これらは上記ステップＳ２２，Ｓ２３で求めた結果を利用できる。なお、添え字の「１」は第１基底〈u_1〉を指すインデクスカウンタｉの内容であり、これは上記ステッ

プS２７の処理により配列［I_k］に保持されている。ステップS４２では（１９）式の演算を行い、結果をレジスタη，κに格納する。

【０１０２】

【数１９】

$$\eta = \frac{1}{\sqrt{L_k}} \qquad \kappa = P_k \eta \tag{19}$$

【０１０３】ステップS４３では後述の内積演算〈u_1・u_i〉の前処理として〈u_1〉の第１６成分を残りの各成分の値から差し引いた１５次元ベクトル〈w_1〉を求める。ステップS４４ではｉ＝０〜（Ｎ－１）につき内積〈w_k・u_i〉ηを求め、これらを配列［Q_i］に格納する。ステップS４５ではｉ＝０〜（Ｎ－１）につき（$P_i - \kappa Q_i$）を求め、これらを配列［P_i］に格納する。ここで、右辺のP_iは上記ステップS２２の演算結果であり、更にこのステップS４５の演算結果をステップS２２の配列［P_i］に上書き保存することで、配列［P_i］の内容は過去の演算結果を反映して逐次更新される。ステップS４６ではｉ＝０〜（Ｎ－１）につき（$L_i - Q_i^2$）を求め、これらを配列［L_i］に格納（上書）する。ここで、右辺のL_iは上記ステップS２３の演算結果であり、更にこのステップS４６の演算結果をステップS２３の配列［L_i］に上書き保存することで、配列［L_i］の内容も過去の演算結果を反映して逐次更新される。以上によるh_iの繰り返し演算は最終的に（２０）式で表せる。

【０１０４】

【数２０】

$$h_i = \frac{(P_i - \kappa Q_i)^2}{L_i - Q_i^2} = \frac{P_i^2}{L_i} \tag{20}$$

【０１０５】ステップS４７ではh_iの最大値を保持するレジスタＥ＝０，基底候補ベクトル〈u_i〉のインデクスカウンタｉ＝０に夫々初期化し、かつ基底数カウンタｋに＋１する。この時点でｋ＝２となる。

【０１０６】ステップS４８では$h_i = P_i^2 / L_i$を求める。ステップS４９ではh_i＞Ｅか否かを判別する。h_i＞Ｅの場合はステップS５０でＥをh_iで更新し、その時のｉを配列［I_k］（ｋ＝２）に保持する。またh_i＞Ｅでない場合は上記ステップS５０の処理をスキップする。

【０１０７】ステップS５１ではｉに＋１し、更にステップS５２ではｉ≧Ｎか否かを判別する。ｉ≧Ｎでない場合はステップS４８に戻り、次のh_iにつき上記同様の最大値探索処理を行う。以下、同様にして進み、やがて、ｉ≧Ｎになると全ネストブロックの探索終了である。この時、上記配列［I_k］（ｋ＝２）にはh_iを最大とするような第２基底ベクトル〈u_2〉のインデクス値ｉが保持されている。

【０１０８】ステップS５３では第２基底ベクトル〈u_2〉を〈v_1〉に正規直交化して正規化基底ベクトル〈v_2〉となし、これを配列［V_k］（ｋ＝２）に格納する。またスカラー係数α_2（〈d_1〉の〈v_2〉への射影）を求め、これを配列［A_k］（ｋ＝２）に格納する。このように基底ベクトル〈u_2〉の正規直交化とスカラー係数α_2の演算は上記探索結果について１回行えばよく、これによってＡＯＴ処理の大幅な軽量化と高速化が図られる。以下も同様である。

【０１０９】ステップS５４では残差ベクトル〈d_1〉を第２基底で近似後の残差ベクトル〈d_2〉＝〈d_1〉－α_2〈v_2〉により更新する。ステップS５５では新たな残差ベクトルの二乗ノルム$e = \| d_2 \|^2$を求め、更にステップS５６ではｅ≦Ｚか否かを判別する。ｅ≦Ｚの場合はこの段階でＡＯＴ処理を終了し、またｅ≦Ｚでない場合は第３基底の探索処理を行う。

【０１１０】図６は第３基底の探索処理を示している。この処理の説明前に該処理を効率よく行うために行った計算上の工夫を説明する。即ち、通常なら第３基底は残差ベクトル〈d_2〉との差の二乗ノルムe_iを最小とする直交ベクトル〈u_i'〉として求められるが、この関係式を更に展開すると（２１）式が得られる。

【０１１１】

【数２１】

$$\begin{aligned} e_i &= \left\| \mathbf{d}_2 - \frac{\langle \mathbf{d}_2 \bullet \mathbf{u}_i' \rangle}{\|\mathbf{u}_i'\|^2} \mathbf{u}_i' \right\|^2 \\ &= \|\mathbf{d}_2\|^2 - 2\frac{\langle \mathbf{d}_2 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} + \frac{\langle \mathbf{d}_2 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^4}\|\mathbf{u}_i'\|^2 \\ &= \|\mathbf{d}_2\|^2 - \frac{\langle \mathbf{d}_2 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} \end{aligned} \tag{21}$$

【０１１２】ここで、直交ベクトル〈u_i'〉は第３基底の候補ベクトル〈u_i〉を第１，第２の正規化基底ベクトル〈v_1〉，〈v_2〉に直交化したものである。

【０１１３】

【数２２】

$$\mathbf{u}_i' = \mathbf{u}_i - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{u}_i \bullet \mathbf{v}_2 \rangle \mathbf{v}_2 \quad (22)$$

【０１１４】同様にして上記（２１）式右辺第１項の$\|d_2\|^2>0$は基底候補によらず一定であるから、同右辺第２項を最大にするような直交ベクトル〈u_i'〉が第３基底となり得る。この右辺第２項をh_iとおく。

【０１１５】

【数２３】

$$h_i = \frac{\langle \mathbf{d}_2 \bullet \mathbf{u}_i' \rangle^2}{\|\mathbf{u}_i'\|^2} \quad (23)$$

【０１１６】更にh_i分子の直交ベクトル〈u_i'〉を基底候補ベクトル〈u_i〉で表すと、h_i分子は（２４）式で表せる。

【０１１７】

【数２４】

$$\begin{aligned}\langle \mathbf{d}_2 \bullet \mathbf{u}_i' \rangle^2 &= \langle \mathbf{d}_2 \bullet (\mathbf{u}_i - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{u}_i \bullet \mathbf{v}_2 \rangle \mathbf{v}_2) \rangle^2 \\ &= (\langle \mathbf{d}_2 \bullet \mathbf{u}_i \rangle - \langle \mathbf{d}_2 \bullet \mathbf{v}_1 \rangle \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle - \langle \mathbf{d}_2 \bullet \mathbf{v}_2 \rangle \langle \mathbf{u}_i \bullet \mathbf{v}_2 \rangle)^2 \\ &= \langle \mathbf{d}_2 \bullet \mathbf{u}_i \rangle^2 \qquad \because \langle \mathbf{d}_2 \bullet \mathbf{v}_1 \rangle = 0 \quad \langle \mathbf{d}_2 \bullet \mathbf{v}_2 \rangle = 0 \end{aligned} \quad (24)$$

【０１１８】更に（２４）式の残差ベクトル〈d_2〉を最初の残差ベクトル〈d〉で表すと、h_i分子は（２５）式で表せる。

【０１１９】

【数２５】

$$\begin{aligned}\langle \mathbf{d}_2 \bullet \mathbf{u}_i \rangle^2 &= \langle (\mathbf{d} - \langle \mathbf{d} \bullet \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{d} \bullet \mathbf{v}_2 \rangle \mathbf{v}_2) \bullet \mathbf{u}_i \rangle^2 \\ &= (\langle \mathbf{d} \bullet \mathbf{u}_i \rangle - \langle \mathbf{d} \bullet \mathbf{v}_1 \rangle \langle \mathbf{v}_1 \bullet \mathbf{u}_i \rangle - \langle \mathbf{d} \bullet \mathbf{v}_2 \rangle \langle \mathbf{v}_2 \bullet \mathbf{u}_i \rangle)^2 \\ &= \left(\langle \mathbf{d} \bullet \mathbf{u}_i \rangle - \frac{\langle \mathbf{d} \bullet \mathbf{u}_1 \rangle}{\|\mathbf{u}_1\|} \frac{\langle \mathbf{u}_1 \bullet \mathbf{u}_i \rangle}{\|\mathbf{u}_1\|} - \frac{\langle \mathbf{d} \bullet \mathbf{u}_2' \rangle}{\|\mathbf{u}_2'\|} \frac{\langle \mathbf{u}_2' \bullet \mathbf{u}_i \rangle}{\|\mathbf{u}_2'\|} \right)^2 \end{aligned} \quad (25)$$

【０１２０】同様にしてh_i分母を変形すると（２６）式で表せる。

【０１２１】

【数２６】

$$\begin{aligned}\|\mathbf{u}_i'\|^2 &= \|\mathbf{u}_i - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{u}_i \bullet \mathbf{v}_2 \rangle \mathbf{v}_2\| \, \|\mathbf{u}_i - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle \mathbf{v}_1 - \langle \mathbf{u}_i \bullet \mathbf{v}_2 \rangle \mathbf{v}_2\| \\ &= \|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle^2 - \langle \mathbf{u}_i \bullet \mathbf{v}_2 \rangle^2 \end{aligned} \quad (26)$$

【０１２２】以上を（２３）式のh_iに代入すると（２７）式が得られる。

【０１２３】

【数２７】

$$h_i = \frac{(\langle \mathbf{d} \bullet \mathbf{u}_i \rangle - \langle \mathbf{d} \bullet \mathbf{v}_1 \rangle \langle \mathbf{v}_1 \bullet \mathbf{u}_i \rangle - \langle \mathbf{d} \bullet \mathbf{v}_2 \rangle \langle \mathbf{v}_2 \bullet \mathbf{u}_i \rangle)^2}{\|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle^2 - \langle \mathbf{u}_i \bullet \mathbf{v}_2 \rangle^2}$$

【０１２４】ところで、（２７）式分子／分母の各第２項までは既に計算されており、これらには（２８）式の関係がある。

【０１２５】

【数２８】

$$P_i = \langle \mathbf{d} \bullet \mathbf{u}_i \rangle - \langle \mathbf{d} \bullet \mathbf{v}_1 \rangle \langle \mathbf{v}_1 \bullet \mathbf{u}_i \rangle \quad (28\text{-}1)$$

$$L_i = \|\mathbf{u}_i\|^2 - \langle \mathbf{u}_i \bullet \mathbf{v}_1 \rangle^2 \quad (28\text{-}2)$$

【０１２６】従って、h_iの演算は上記（１８－２）式に習って最終的に（２９）式で表せる。

【０１２７】

【数２９】

$$h_i = \frac{\left(P_i - \frac{P_k}{\sqrt{L_k}} \frac{\langle \mathbf{v}_k \bullet \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2}{L_i - \left(\frac{\langle \mathbf{v}_k \bullet \mathbf{u}_i \rangle}{\sqrt{L_k}} \right)^2} \quad (29)$$

【０１２８】（２９）式は内積〈u_k・u_i〉が〈v_k・u_i〉になっていることを除き、上記（１８－２）式と同じ形をしている。従って、これ以降の各基底は図５と同様のルーティンを再帰的に使用することで効率よく求まる。

【０１２９】以上を前提として、第３基底以降の探索は以下の演算処理を行う。即ち、ステップＳ６１ではｋ＝２によりP_2＝〈d_1・u_2〉，L_2＝$\|u_2\|^2$を保持す

る。ステップＳ６２では（３０）式の演算を行い、結果をレジスタη，κに格納する。

【０１３０】

【数３０】

$$\eta=\frac{1}{\sqrt{L_k}}\qquad \kappa=P_k\eta \tag{30}$$

【０１３１】ステップＳ６３では後述の内積演算〈v_2・u_i〉の前処理として〈v_2〉の第１６成分を残りの各成分の値から差し引いた１５次元ベクトル〈w_2〉を求める。但し、〈v_2〉の各成分は整数値ではないので、このままでは内積演算を実数型で行う必要が生じる。これを避けるために、予め〈v_2〉（即ち、〈w_2〉）の各成分に定数ａを掛けて整数にしておく。

【０１３２】ステップＳ６４ではｉ＝０～（Ｎ－１）につき内積（〈w_2・u_i〉η／ａ）を求め、これらを配列［Q_i］に格納（上書）する。この時、各演算結果を定数ａで割ることにより、位を戻す。ステップＳ６５ではｉ＝０～（Ｎ－１）につき（$P_i-\kappa Q_i$）を求め、これらを配列［P_i］に格納（上書）する。ステップＳ６６ではｉ＝０～（Ｎ－１）につき（$L_i-Q_i^2$）を求め、これらを配列［L_i］に格納（上書）する。以上により上記（２９）式の演算は（３１）式で表せる。

【０１３３】

【数３１】

$$h_i=\frac{(P_i-\kappa Q_i)^2}{L_i-Q_i^2}=\frac{P_i^2}{L_i} \tag{31}$$

【０１３４】ステップＳ６７ではh_iの最大値を保持するレジスタＥ＝０，基底候補ベクトル〈u_i〉のインデクスカウンタｉ＝０に夫々初期化し、かつ基底数カウンタｋに＋１する。この時点でｋ＝３となる。

【０１３５】ステップＳ６８では$h_i=P_i^2/L_i$を求める。ステップＳ６９ではh_i＞Ｅか否かを判別する。h_i＞Ｅの場合はステップＳ７０でＥをh_iで更新し、その時のｉを配列［I_k］（ｋ＝３）に保持する。またh_i＞Ｅでない場合は上記ステップＳ７０の処理をスキップする。

【０１３６】ステップＳ７１ではｉに＋１し、更にステップＳ７２ではｉ≧Ｎか否かを判別する。ｉ≧Ｎでない場合はステップＳ６８に戻り、次のh_iにつき上記同様の最大値探索処理を行う。以下、同様にして進み、やがて、ｉ≧Ｎになると全ネストブロックの探索終了である。この時、上記配列［I_k］（ｋ＝３）にはh_iを最大とするような第３基底ベクトル〈u_3〉のインデクス値ｉが保持されている。

【０１３７】ステップＳ７３では第３基底ベクトル〈u_3〉を〈v_1〉，〈v_2〉に直交化かつ正規化して正規化基底ベクトル〈v_3〉となし、これを配列［V_k］に格納する。またスカラー係数α_3（〈d_2〉の〈v_3〉への射影）を求め、これを配列［A_k］に格納する。

【０１３８】ステップＳ７４では残差ベクトル〈d_2〉を第３基底で近似後の残差ベクトル〈d_3〉＝〈d_2〉－α_3〈v_3〉により更新する。ステップＳ７５では新たな残差ベクトルの二乗ノルム$e=\|d_3\|^2$を求め、更にステップＳ７６ではｅ≦Ｚか否かを判別する。ｅ≦Ｚの場合はこの段階でＡＯＴ処理を終了し、またｅ≦Ｚでない場合はステップＳ６１に戻り、第４基底以降の前処理及び探索処理を行う。なお、図示しないが、好ましくは、例えば上記ステップＳ７６の次にｋ≧４か否かの判別処理を設け、ｋ≧４の場合はこのＡＯＴ処理を抜けるようにする。

【０１３９】以上により、ＡＯＴ処理の大幅な軽量化、かつ高速化が可能となり、演算時間は実測比で従来の１／３～１／１０に短縮された。

【０１４０】図２を参照し、ＡＯＴ３２からはα_k，〈v_k〉（ｋ＝１～ｍ）の組が得られ、これらの一次結合により残差ベクトル〈d_j〉を許容誤差Ｚ以内に近似できる。更に、係数変換部３３はα_k，〈v_k〉（ｋ＝１～ｍ）の組をβ_k，〈u_k〉（ｋ＝１～ｍ）の組に変換すべく、以下（従来と同様）の方法により展開係数β_kを求める。即ち、今、基底候補ベクトル〈u_k〉，展開係数β_k，正規化基底ベクトル〈v_k〉，スカラー係数α_kの各行列を（３２）式とおく時に、

【０１４１】

【数３２】

$$\mathbf{U}=[\mathbf{u}_1,\mathbf{u}_2,\cdots\mathbf{u}_m]\qquad \mathbf{B}=\begin{bmatrix}\beta_1\\ \beta_2\\ \vdots\\ \beta_m\end{bmatrix}$$

$$\mathbf{V}=[\mathbf{v}_1,\mathbf{v}_2,\cdots\mathbf{v}_m]\qquad \mathbf{A}=\begin{bmatrix}\alpha_1\\ \alpha_2\\ \vdots\\ \alpha_m\end{bmatrix} \tag{32}$$

【０１４２】これらを（３３）式で関係つける。

【０１４３】

【数33】

$$\mathbf{UB} = \mathbf{VA} \qquad (33)$$

【0144】これを行列Bについて解くためには、まず行列Uを正方行列に変換すべく、両辺に行列Uの転置行列U^Tを左側から掛ける。

【0145】

【数34】

$$\mathbf{U}^T\mathbf{UB} = \mathbf{U}^T\mathbf{VA} \qquad (34)$$

【0146】この行列（U^TU）は、（35）式の様に展開され、

【0147】

【数35】

$$\mathbf{U}^T\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_{nk} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1, \mathbf{u}_2, \cdots \mathbf{u}_{nk} \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{u}_1 \bullet \mathbf{u}_1 & \mathbf{u}_1 \bullet \mathbf{u}_2 & \cdots & \mathbf{u}_1 \bullet \mathbf{u}_{nk} \\ \mathbf{u}_2 \bullet \mathbf{u}_1 & \mathbf{u}_2 \bullet \mathbf{u}_2 & \cdots & \mathbf{u}_2 \bullet \mathbf{u}_{nk} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{u}_{nk} \bullet \mathbf{u}_1 & \mathbf{u}_{nk} \bullet \mathbf{u}_2 & \cdots & \mathbf{u}_{nk} \bullet \mathbf{u}_{nk} \end{bmatrix} \qquad (35)$$

【0148】ここで〈u_i・u_j〉は内積を表し、かつ〈u_i・u_j〉＝〈u_j・u_i〉であるから、対角要素に対して対称な正方行列が得られ、かつ〈u_i〉と〈u_j〉とが異なるから、逆行列が存在する。そこで、更に両辺の左側から行列（U^TU）の逆行列（U^TU）$^{-1}$を掛けることで（36）式が得られ、β_kが求まる。

【0149】

【数36】

$$\left(\mathbf{U}^T\mathbf{U}\right)^{-1}\mathbf{U}^T\mathbf{UB} = \mathbf{B} = \left(\mathbf{U}^T\mathbf{U}\right)^{-1}\mathbf{U}^T\mathbf{VA} \qquad (36)$$

【0150】このように正規直交基底系α_k，〈v_k〉（k＝1～m）の組を非直交基底系β_k，〈u_k〉（k＝1～m）の組に変換することにより、復号側では各基底候補ベクトル〈u_k〉を一々直交化する必要は無く、夫々にβ_kを掛けて加算することにより残差ベクトル〈d_j〉を近似できる。従って、復号処理を簡単かつ高速に行える。次に展開係数β_kの圧縮符号処理を説明する。

【0151】図9は実施の形態による展開係数符号処理のイメージ図である。図9（a）において、生成されたβ_1～β_4からノルム（大きさ）を抽出する。図9（b）において、ノルムを例えば昇順（β_3，β_2，β_4，β_1）に並べ換え、前方（最初は0）から順に差分（$\Delta\beta_3$，$\Delta\beta_2$，$\Delta\beta_4$，$\Delta\beta_1$）を求める。図9（c）において、係数残差（$\Delta\beta_3$，$\Delta\beta_2$，$\Delta\beta_4$，$\Delta\beta_1$）をその下位2ビットと上位ビットとに分離し、上位ビットをハフマン符号化する。

【0152】この例では$\Delta\beta_3$と（$\Delta\beta_2$＝$\Delta\beta_4$＝$\Delta\beta_1$）との2組の値が発生しており、よってハフマン符号では発生頻度の高い（$\Delta\beta_2$，$\Delta\beta_4$，$\Delta\beta_1$）にはビット数の少ない符号が割り振られ、また発生頻度の低い$\Delta\beta_3$にはビット数の多い符号が割り振られる。従って、展開係数β_kの圧縮符号化が可能となる。しかも、係数残差$\Delta\beta_k$の上位ビットをハフマン符号化する構成により、下位ビットの端数分が切り離されることになり、よって上位ビットでは図示の如く$\Delta\beta_2$＝$\Delta\beta_4$＝$\Delta\beta_1$となるような可能性が高い。

【0153】また、残差$\Delta\beta_k$の下位2ビットは正負の符号ビットと共に対応する基底ベクトル〈u_k〉のインデクス情報（13ビット＝0～8191）と共に2バイト固定長符号エリアにパッキングされ、固定長符号として出力される。これらの符号の出力順は$\Delta\beta_3$，$\Delta\beta_2$，$\Delta\beta_4$，$\Delta\beta_1$（即ち、u_3，u_2，u_4，u_1）の順である。

【0154】図9（d）において、復号側では各符号をu_3，u_2，u_4，u_1の順で入力し、夫々から係数$\Delta\beta_3$，$\Delta\beta_2$，$\Delta\beta_4$，$\Delta\beta_1$を分離する。更に最初の$\Delta\beta_3$からβ_3を復号し、該β_3に$\Delta\beta_2$を加えてβ_2を復号し、該β_2に$\Delta\beta_4$を加えてβ_4を復号し、そして、該β_4に$\Delta\beta_1$を加えてβ_1を復号する。β_k〈u_k〉はこれらの和（一次結合）をとって機能するものであるから、これらの順序は問題ではない。

【0155】なお、上記ノルムを昇順に並べ換え、前方（最初は0）から順に差分を求めたが、逆にノルムを降順に並べ換え、後方（最初は0）から順に差分を求めても良い。

【0156】以下、符号部34による符号処理を説明する。DPCMの予測残差$\Delta DC_{J,I}$については量子化係数Q（Z）で量子化すると共に、$\Delta DC_{J,I}$＝0の場合のみランレングスを考慮し、予測残差$\Delta DC_{J,I}$及びランレングスを夫々独立にハフマン符号化する。基底数kは、k＝0の場合のみランレングスを考慮し、基底数k及びランレングスを夫々独立にハフマン符号化する。

係数残差$\Delta\beta_k$の上位ビットは定数Q（例えば8）で量子化した商をハフマン符号化する。また基底ベクトル〈u_k〉のコード情報ｉ（＝１３ビット）に展開係数β_kの符号ビット及び係数残差$\Delta\beta_k$の下位２ビットを詰めて計１６ビットの固定長符号となし、これらは残差$\Delta\beta_k$の昇順（又は降順）に詰めて送られる。全体としては画素ブロック単位で出現順に詰めて符号列を構成する。必要なら画素ブロックの切り替わりを示すための符号ＥＯＢを書き込む。

【０１５７】図１０は実施の形態による画像復号装置のブロック図で、上記図２の画像符号装置に対応したものである。図において、４１はハフマン等による復号部、４２は注目画素DC_Jを含む周囲のＤＣ値DC_J'から交流成分を含むターゲットブロック〈R_j〉を推定する交流成分予測部、４３は復号基底系β_k〈u_k〉（ｋ＝１～ｍ）に基づきく近似残差ベクトル〈d_j〉を再生する残差ベクトル再生部、４４は復号ブロック〈R_j〉に基づきターゲットブロック〈R_j〉を再生するR_j再生部、４５は再生画像を記憶する再生画像メモリ、４６は復号ＤＣ値をＩＤＰＣＭ復号するＩＤＰＣＭ部、４７は復号ＤＣ画像を記憶するＤＣ画像メモリ、４８は図２と同様のＤＣネスト生成部、４９はＤＣネストを記憶するＤＣネストメモリ、５０はＤＣネストからダウンサンプルされた選択ブロック〈U_k〉を保持する選択ブロックバッファ、５１は〈U_k〉にβ_kを乗算する乗算器、５２，５３はβ_k〈U_k〉（ｋ＝１～ｍ）の累積加算部、５４は累積加算結果のブロック平均値A_jを求める平均器、５５は累積加算結果からブロック平均値A_jを分離する減算器、５６は再生近似残差ベクトル〈d_j〉を保持する近似ベクトルバッファ、５７はターゲットブロック〈R_j〉の再生ＤＣ値DC_Jに再生近似残差ベクトル〈d_j〉を加算する加算器である。

【０１５８】図１１は実施の形態による画像復号処理のフローチャートである。ステップＳ１０１では画像符号データを読み込む。ステップＳ１０２では図２と同様のＩＤＰＣＭ法によりＹ，Ｕ，Ｖの各ＤＣ値を解凍（復号）し、ＤＣ画像を再生する。ステップＳ１０３ではＹ成分のＤＣ画像からＤＣネストを生成する。この時、上記図７で示した如く、各ＤＣ画素値DC_Jの下位４ビットがマスク（＝０）され、各ＤＣネスト画素値N_jとなる。なお、ＤＣ画像の切り出し位置等の情報は別途に受け取る。ステップＳ１０４では原画像メモリ４５及びＤＣ画像メモリ４７に対するインデクスカウンタｊ，Ｊを共に０に初期化する。

【０１５９】ステップＳ１０５では１ブロック画像分の符号データを入力する。ステップＳ１０６では基底数ｋ＝０か否かを判別する。ｋ＝０の場合はステップＳ１１４で後述する交流成分予測法によりターゲットブロック〈R_j〉を再生する。またｋ≠０の場合は更にステップＳ１０７で１≦ｋ≦４か否かを判別する。

【０１６０】１≦ｋ≦４の場合はステップＳ１１２で残差ベクトル〈d_j〉を逆量子化する。本実施の形態では予めＤＣネストの下位４ビットがマスク（＝０）されているため、各選択ブロック〈U_k〉に直接β_kを掛けてこれらを累積加算し、累積加算結果からそのブロック平均値A_jを１回だけ分離することで残差ベクトル〈d_j〉が一挙に得られる。よって復号処理が高速化される。ステップＳ１１３では得られた残差ベクトル〈d_j〉に対応するＤＣ値DC_Jを加算する。

【０１６１】また１≦ｋ≦４でない場合はステップＳ１０８でターゲットブロック〈R_j〉の復号データよりターゲットブロック〈R_j〉を直接再生する。こうして、上記何れかの方法により４×４画素のターゲットブロック〈R_j〉が再生された。ステップＳ１０９では再生されたターゲットブロック〈R_j〉を再生画像メモリ４５に格納する。

【０１６２】ステップＳ１１０ではカウンタｊ，Ｊに夫々＋１し、更にステップＳ１１１ではｉ≧Ｍ（全画素ブロック数）か否かを判別する。ｉ≧Ｍでない場合はステップＳ１０５に戻り、次のブロック画像符号データにつき上記同様の復号・再生処理を行う。以下同様にして進み、やがて、ステップＳ１１１の判別でｊ≧Ｍになると、１画像分の復号処理を終了する。

【０１６３】図１２は実施の形態における交流成分予測のイメージ図で、公知の予測法を採用できる．図１２（Ａ）は段階的交流成分予測法を示しており、以下に内容を概説する。その第１段階では注目ブロックＳ上の各サブブロックS_1～S_4を該Ｓを含む周囲４ブロック（Ｕ，Ｒ，Ｂ，Ｌ）の各ＤＣ値から次式により推定する。

【０１６４】

$S_1 = S + (U + L - B - R) / 8$

$S_2 = S + (U + R - B - L) / 8$

$S_3 = S + (B + L - U - R) / 8$

$S_4 = S + (B + R - U - L) / 8$

同様にして、この第１段階目ではU_1～U_4，L_1～L_4，R_1～R_4，B_1～B_4等が推定される。更に、その第２段階では上記方法を再帰的に使用することで、S_1上の４画素P_1～P_4を次式により推定する。

【０１６５】

$P_1 = S_1 + (U_3 + L_2 - S_3 - S_2) / 8$

$P_2 = S_1 + (U_3 + S_2 - S_3 - L_2) / 8$

$P_3 = S_1 + (S_3 + L_2 - U_3 - S_2) / 8$

$P_4 = S_1 + (S_3 + S_2 - U_3 - L_2) / 8$

S_2～S_4上の各４画素P_1～P_4についても同様である。このような２段階処理によりターゲットブロック〈R_j〉が再生される。

【０１６６】図１３（Ｂ）は本件出願人による既提案の非段階的交流成分予測法を示しており、注目ブロックＳを含む周囲４ブロック（Ｕ，Ｒ，Ｂ，Ｌ）の各ＤＣ値か

ら各サブブロックS_1～S_4における各４画素P_1～P_4を一挙に推定する。以下内容を概説する。まずS_1上の４画素P_1～P_4を求める場合は、$S_2 \fallingdotseq S_3 \fallingdotseq S$，$U_3 \fallingdotseq U$，$L_2 \fallingdotseq L$の各近似を行う。この近似を上記$S_1$上の$P_1$の式に適用すると、

$$P_1 = S_1 + (U_3 + L_2 - S_3 - S_2)/8$$
$$= S_1 + (U + L - S - S)/8$$

が得られる。更にこの式に上記S_1の式、$S_1 = S + (U + L - B - R)/8$を代入すると、$S_1$上の$P_1$は最終的に、

$$P_1 = S + (2U + 2L - 2S - B - R)/8$$

で表せる。また上記S_1上のP_2については、

$$P_2 = S_1 + (U_3 + S_2 - S_3 - L_2)/8$$
$$= S_1 + (U + S - S - L)/8$$

が得られる。更にこの式に上記S_1の式、$S_1 = S + (U + L - B - R)/8$を代入すると、$S_1$上の$P_2$は最終的に、

$$P_2 = S + (2U - B - R)/8$$

で表せる。また上記S_1上のP_3については、

$$P_3 = S_1 + (S_3 + L_2 - U_3 - S_2)/8$$
$$= S_1 + (S + L - U - S)/8$$

が得られる。更にこの式に上記S_1の式、$S_1 = S + (U + L - B - R)/8$を代入すると、$S_1$上の$P_3$は最終的に、

$$P_3 = S + (2L - B - R)/8$$

で表せる。また上記S_1上のP_4については、

$$P_4 = S_1 + (S_3 + S_2 - U_3 - L_2)/8$$
$$= S_1 + (S + S - U - L)/8$$

が得られる。更にこの式に上記S_1の式、$S_1 = S + (U + L - B - R)/8$を代入すると、$S_1$上の$P_4$は最終的に、

$$P_4 = S + (2S - B - R)/8$$

で表せる。従って、S_1上の４画素P_1～P_4は、

$$P_1 = S + (2U + 2L - 2S - B - R)/8$$
$$P_2 = S + (2U - B - R)/8$$
$$P_3 = S + (2L - B - R)/8$$
$$P_4 = S + (2S - B - R)/8$$

により非段階的に一挙に求まる。S_2～S_4上の各４画素P_1～P_4についても同様である。

【０１６７】なお、上記実施の形態を具体的数値例を伴って説明したが本発明がこれらに限定されないことは明らかである。

【０１６８】また、上記本発明に好適なる実施の形態を述べたが、本発明思想を逸脱しない範囲内で各部の構成、制御、処理及びこれらの組合せの様々な変更が行えることは言うまでも無い。

【０１６９】

【発明の効果】以上述べた如く本発明によれば、ＤＣネストの改良により高画質が得られ、またＡＯＴ演算の工夫により高速符号化が得られた。従って、ＨＶＱ方式の高画質化、高速符号化に寄与するところが極めて大きい。

【図面の簡単な説明】

【図１】本発明の原理を説明する図である。

【図２】実施の形態による画像符号装置のブロック図である。

【図３】実施の形態による画像符号（メイン）処理のフローチャートである。

【図４】実施の形態による適応的直交変換処理のフローチャート（１）である。

【図５】実施の形態による適応的直交変換処理のフローチャート（２）である。

【図６】実施の形態による適応的直交変換処理のフローチャート（３）である。

【図７】実施の形態によるＤＣネストの説明図（１）である。

【図８】実施の形態によるＤＣネストの説明図（２）である。

【図９】実施の形態による展開係数符号処理のイメージ図である。

【図１０】実施の形態による画像復号装置のブロック図である。

【図１１】実施の形態による画像復号処理のフローチャートである。

【図１２】実施の形態における交流成分予測のイメージ図である。

【図１３】従来の画像符号装置のブロック図である。

【図１４】従来の適応的直交変換処理のフローチャートである。

【図１５】従来の適応的直交変換処理のイメージ図である。

【図１６】従来の平均値分離処理のイメージ図である。

【符号の説明】

１１　原画像メモリ
１２　ＤＣ値生成部
１３　差分ＰＣＭ符号部（ＤＰＣＭ）
１４　逆ＤＰＣＭ符号部（ＩＤＰＣＭ）
１５　ＤＣ画像メモリ
１６　ＤＣネスト生成部
１７　ＤＣネストメモリ
１８　減算器
１９　残差ベクトルバッファ
２０　抽出部ブロックバッファ
２１　平均器
２２　減算器
２３　候補ベクトルバッファ
２４　適応的直交変換処理部（ＡＯＴ）
２５　係数変換部
２６　符号部
３１　ＤＣネスト生成部

３２　適応的直交変換処理部（ＡＯＴ）
３３　係数変換部
３４　符号部

【図１】

本発明の原理を説明する図